Naval Research Laboratory

Washington, DC 20375-5000

Filtering Interpolators

ROBERT L. LUCKE

Advanced Concepts Branch
Optical Sciences Division

and

ALAN D. STOCKER

Space Computer Corporation
Suite 104
2800 Olympic Boulevard
Santa Monica, CA 90404-4119

July 20, 1992

REPORT DOCUMENTATION PAGE

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188), Washington, DC 20503.

1. AGENCY USE ONLY (Leave blank)	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	July 20, 1992	Final Report 6/91-6/92

4. TITLE AND SUBTITLE	5. FUNDING NUMBERS
Filtering Interpolators	

6. AUTHOR(S)

Robert L. Lucke and Alan D. Stocker*

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)	8. PERFORMING ORGANIZATION REPORT NUMBER
Naval Research Laboratory Washington, DC 20375-5000	NRL/FR/6521—92-9505

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSORING/MONITORING AGENCY REPORT NUMBER
Office of the Chief of Naval Research Arlington, VA 22217-5000	

11. SUPPLEMENTARY NOTES

*Space Computer Corporation, Santa Monica, CA 90404-4119

12a. DISTRIBUTION/AVAILABILITY STATEMENT	12b. DISTRIBUTION CODE
Approved for public release; distribution unlimited.	

13. ABSTRACT (Maximum 200 words)

Interpolation error contributes to clutter leakage through a frame differencing signal processor, especially for highly structured scenes. This error can be reduced by an order of magnitude by applying a spectral filter to the unshifted frame that matches the filtering effect of applying the interpolator to the frame that is shifted. The signature reduction penalty for point targets is rarely as large as a factor of two, leading to substantial improvements in signal-to-clutter ratios when interpolation error is the dominant source of clutter. Parameterized families of local convolutional interpolators (polynomial and trigonometric) that can be adjusted to the particular target/clutter/noise combination of interest are presented. For spline interpolation applications, the trigonometric family yields an alternative to the cubic B-spline kernel.

14. SUBJECT TERMS		15. NUMBER OF PAGES
Interpolation Filtering Frame differencing		26
		16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
UNCLASSIFIED	UNCLASSIFIED	UNCLASSIFIED	UL

CONTENTS

FILTERING INTERPOLATORS

I. INTRODUCTION

In many electro-optical imaging systems, the most serious impediment to detecting low-contrast moving targets is spatially varying background clutter. Multiframe clutter-suppression processing can often provide an effective solution to this problem. In its simplest form, this processing involves subtracting a second image from a first in a frame-differencing signal processor (FDSP), in order to suppress the fixed background on a pixel-by-pixel basis. (The emphasis here will be on a first-order FDSP, but the interpolators described will also be applicable to higher-order FDSPs, or to any other signal processor that compares multiple frames of data.) Since one frame must almost always be resampled before it is compared to the other, interpolation error contributes to clutter leakage in the difference frame. The key idea of this report is to apply a spectral filter to the unshifted data that matches the filtering effect of applying the interpolator to the data that are shifted, thereby minimizing the effect of interpolation error in the difference frame. Theoretical analysis indicates that the result, which we will call a filtering interpolator, gives large clutter leakage reductions with minimal loss of target signature. Tests on simulated imagery confirm target-to-clutter ratio improvements of a factor of five or more for those highly structured scenes examined, in the absence of other sources of error. (The simulation assumes a space-based IR surveillance sensor, but that is not important to the results presented here.)

To set up the mathematical problem, let x_n denote the value at $u = n$ of data taken from a continuous function $f(u)$, (i.e., $x_n = f(n)$). Consider two sets of equally spaced samples $\{x_n\}$ and $\{x_{n+s}\}$, as shown in Fig. 1, the second set being taken at points shifted to the right a distance s from the first. Since integral-sample shifts are effected simply by reindexing the data set and do not require interpolation, we will assume $0 \leq s \leq 1$. The first data set $\{x_n\}$ will be called the shift data, because it must be processed to find an interpolated value; the second set $\{x_{n+s}\}$ will be called the no-shift data. Interpolations are always done between the two central points: a new value y_s must be generated from the $\{x_n\}$ to compare to x_s. (The set $\{x_n\}$ is then moved over one sample for the next interpolation.) The new value y_s is to be given by a local formula (as opposed to a global spline method [1,2]):

$$y_s = \sum_{n=-(N/2-1)}^{N/2} x_n A_n(s),$$

(1)

where the $A_n(s)$, the interpolation coefficients, are functions of the shift.

We define what will be called standard interpolators as those for which $A_n(0) = \delta(n)$; that is, $A_0(0) = 1$, $A_n(0) = 0$ for $n \neq 0$. When the shift is zero, these interpolators simply reproduce the data to which they are applied: $y_s = x_s = x_o$. But, for arbitrary data, no interpolator will be perfect for $s \neq 0$; there will be some error when y_s is compared to x_s. Interpolators tend to reproduce some frequency components (usually low) of data with high fidelity, but do less well on other frequency components (usually high). Thus, the shifted data have been through a spectral filter, while the unshifted data have not. The resulting spectral mismatch can lead to significant clutter leakage in the difference frame. While contemplating this problem, one of us [Stocker] realized that if the right spectral filter could be applied to the no-shift data, then the difference between y_s and x_s would be reduced, perhaps dramatically. Of course, such a filter would also reduce high-frequency detail.
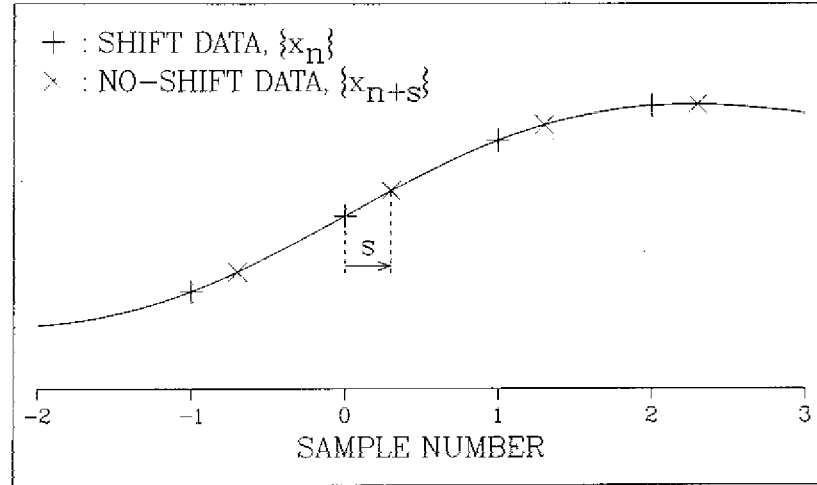

Fig. 1 — The interpolation problem: the solid curve represents the analog signal and the shift and no-shift data sets are separated by the shift s. An interpolated value must be found from the shift data set to compare to the no-shift data value x_s.

In this report, we develop parameterized families of filtering interpolators that permit the user to choose the best trade-off between clutter suppression and high-frequency retention for a particular application. One-dimensional data will be considered explicitly; for two dimensions, the interpolators are applied to each dimension separately.

For our purposes, the shift s is assumed to be known. Usually, s must be determined from the data itself, a problem addressed by Kuglin and Hines [3] and Schaum and McHugh [4]. The shift can also be found by searching for the value of s that minimizes the difference frame [5]. Our ultimate aim is to minimize clutter in an FDSP, so s must be accurately known. A small error δs in the determination of shift will lead to an error in the difference frame of approximately $f'(u)\delta s$.

2. A PRESCRIPTION FOR INTERPOLATORS

One way to generate the A_n is to select a set of basis functions $f_m(x)$ that are thought to fit the data fairly well and require that these functions be interpolated exactly. That is, the A_n are selected to satisfy the equations

$$\sum_{n=-(N/2-1)}^{N/2} f_m(n)A_n(s) = f_m(s), \tag{2}$$

with $m = 1, \ldots, N$. Note that for $s = 0$, Eq. (2) is satisfied by $A_n(0) = \delta(n)$. Solution of these equations leads to standard interpolators for any desired basis functions.

For the polynomial basis functions $f_m(x) = x^{m-1}$, the resulting A's give standard Lagrange interpolation [1], that is, the data are fit with a polynomial of order $N - 1$, from which y, is calculated. The coefficients for these Lagrange functions for N data points (LF-N) interpolators are, for even N,

$$A_n(s) = \prod_{\substack{k=-(N/2-1) \\ k \neq 0}}^{N/2} \frac{(s-k)}{(n-k)}. \tag{3}$$

2

Another possible basis for interpolation is the set of trigonometric functions used by the discrete Fourier transform (DFT). For $N = 4$, these are

$$
\begin{aligned}
f_1(x) &= 1, \\
f_2(x) &= \cos(\pi x/2), \\
f_3(x) &= \sin(\pi x/2), \\
f_4(x) &= \cos(\pi x).
\end{aligned}
\tag{4}
$$

Substitution of these functions in Eq. (2) yields the DFT-4 interpolator that was shown by Lucke [6] to give better performance than LF-4 when the data sampling rate is low.

Interpolators derived from Eq. (2) are of the standard type: when applied with $s = 0$, they simply reproduce the data. We now rewrite Eq. (2) to include the condition that the interpolator be applied to the no-shift data, but only to filter it, not to shift it:

$$
\sum_{n=-(N/2-1)}^{N/2} f_m(n) A_n(s) = \sum_{n=-(N/2-1)}^{N/2} f_m(n+s) A_n(0).
\tag{5}
$$

The left-hand side of Eq. (5) is, as before, the result of applying the interpolator to the shift data. The right-hand side is the result of applying it to the no-shift data. The requirement that the two be equal preserves the exact interpolation of the basis functions in the sense that, when the difference of the two processed data sets is taken, the result is zero. (Filtering interpolators, when applied with $s = 0$, do *not* reproduce the data to which they are applied; that is, from Eq. (1), $y_o \neq x_o$.) Equation (5) allows the $A_n(s)$ to be found for a given set of basis functions once the $A_n(0)$ are chosen. In what follows, we will give forms for arbitrary even N where possible but will usually restrict our attention to $N = 4$. Results for $N = 6$ will be given in Section 7.

We will require that the interpolator reproduce a constant signal exactly (i.e., $f_1(x) = 1$ so that $\Sigma A_n(s) = 1 \; \forall \; s$), and that the no-shift filter be symmetric about $n = 0$ so that its phase is a linear function of frequency. The $A_n(0)$ are thereby determined up to a single parameter a:

$$
\begin{aligned}
A_{-1}(0) &= a, \\
A_0(0) &= 1 - 2a, \\
A_1(0) &= a, \\
A_2(0) &= 0.
\end{aligned}
\tag{6}
$$

Only three samples of the no-shift data set have nonzero coefficients; when $a = 0$, only one sample has a nonzero coefficient and Eq. (5) reduces to Eq. (2).

When the $A_n(0)$ from Eq. (6) are substituted into Eq. (5), along with the LF-4 or DFT-4 basis functions, the resulting equations can be solved for the $A_n(s)$. The interpolation coefficients for LF-4 are

3

$$A_{-1}(s) = \frac{1}{6} [6a - 2(1 + 3a) s + 3s^2 - s^3],$$

$$A_0(s) = \frac{1}{2} [2(1 - 2a) - (1 - 6a) s - 2s^2 + s^3],$$

$$(7)$$

$$A_1(s) = \frac{1}{2} [2a + 2(1 - 3a) s + s^2 - s^3],$$

$$A_2(s) = \frac{1}{6} [- (1 - 6a) s + s^3].$$

For DFT-4 we, find

$$A_{-1}(s) = \frac{1}{4} [1 - 2(1 - 2a)\sin(\pi s/2) - (1 - 4a)\cos(\pi s)],$$

$$A_0(s) = \frac{1}{4} [1 + 2(1 - 2a)\cos(\pi s/2) + (1 - 4a)\cos(\pi s)],$$

$$(8)$$

$$A_1(s) = \frac{1}{4} [1 + 2(1 - 2a)\sin(\pi s/2) - (1 - 4a)\cos(\pi s)],$$

$$A_2(s) = \frac{1}{4} [1 - 2(1 - 2a)\cos(\pi s/2) + (1 - 4a)\cos(\pi s)].$$

At this point, we introduce the interpolation function $h(x)$ defined by the equation

$$h(n - s) = A_n(s),$$ (9)

and $h(x) = 0$ for $|x| > N/2$. The interpolation function $h(x)$ is well-defined because $0 \le s \le 1$. Note that $h(1) = h(-1) = a$ and $h(0) = 1 - 2a$. For DFT-4, the interpolation function is

$$h(x) = \frac{1}{4} [1 + 2(1 - 2a)\cos(\pi x/2) + (1 - 4a)\cos(\pi x)].$$ (10)

With $a = 0$, this is the standard DFT-4 interpolation function given by Lucke [6]. Figure 2 plots the LF-4 and DFT-4 interpolation functions for $a = 0$ (standard interpolation) and for $a = 0.21$ and $a = 0.2$, respectively, (the best values for filtering interpolation, as we see in the next section).

3. OPTIMIZATION

The value of the parameter "a" remains to be determined. The motivation for filtering the no-shift data is to perform the same spectral filtering that the interpolation process performs on the shift data, thereby reducing the difference between the two data sets. In principle, this means matching both the amplitude and phase of the digital filters applied to the two data sets. Here we will concentrate on matching amplitudes, because this leads to a very effective solution to the FDSP clutter suppression problem. We will see later how a considerable degree of phase control is built into Eq. (5) and show in the Appendix how the problem of matching phases of the filters for the two data sets can be approached directly. Detailed examinations of amplitude and phase matching in interpolators and in differencing filters are developed by Schaum [7].
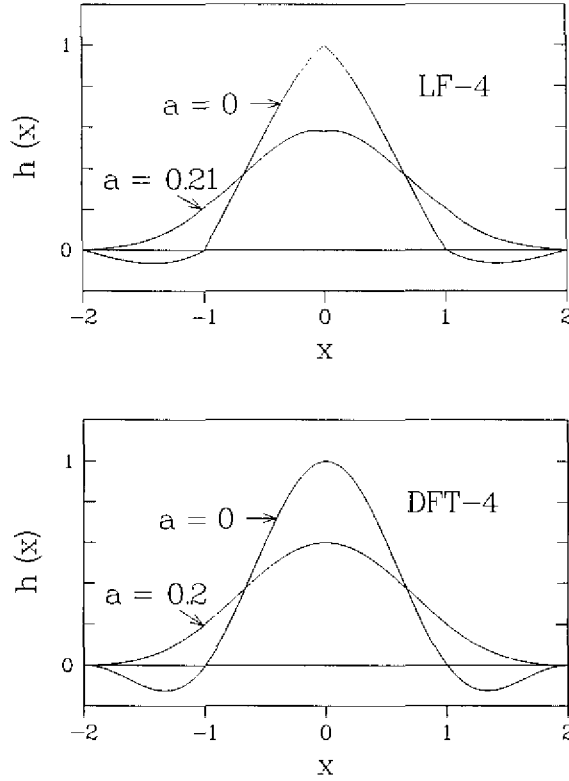
4

Fig. 2 — Interpolation functions for LF-4 and DFT-4, in standard form ($a = 0$) and best-filtering form ($a = 0.21$ and $a = 0.2$)

We see from Eq. (5) that the filtering coefficients applied to the no-shift data are the interpolation coefficients for $s = 0$, while those applied to the shift data can be for any value of s. Therefore, in order to make both sets of coefficients perform (nearly) the same spectral filtering, we seek the value of "a" in the interpolation function that will make the amplitude of spectral filtering as independent of shift as possible. The value of "a" that fulfills this condition will be called the best-filtering "a."

Letting $H(f,s)$, where f is the frequency in cycles/sample, signify the shift-dependent spectral filter for which the tap weights are the A_n, we use standard methods [8] to find the magnitude of $H(f,s)$:

$$|H(f,s)|^2 = \sum_{n=-(N/2-1)}^{N/2} A_n^2(s) + 2 \sum_{m=1}^{N-1} \sum_{n=-(N/2-1)}^{N/2-m} A_n(s)A_{n+m}(s)\cos(2\pi mf). \qquad (11)$$

Figures 3(a) and (b) show contour plots of $|H(f,s)|$ for LF-4 and DFT-4, respectively, in their standard form ($a = 0$). Both can be seen to produce substantial filtering of high frequencies for nonzero shifts; more to the point, the filtering is a strong function of shift.

We now investigate how to choose "a" so that $|H(f,s)|$ is as independent of s as possible. This amounts to making the contours of $|H(f,s)|$ resemble straight lines parallel to the shift axis. With one parameter, this is best done for a fixed frequency $f = f_o$; since most imagery data are dominated by low frequencies, f_o should be near zero. To make a contour line flat, we will require that the derivatives of the function with respect to s vanish to as high an order as possible.
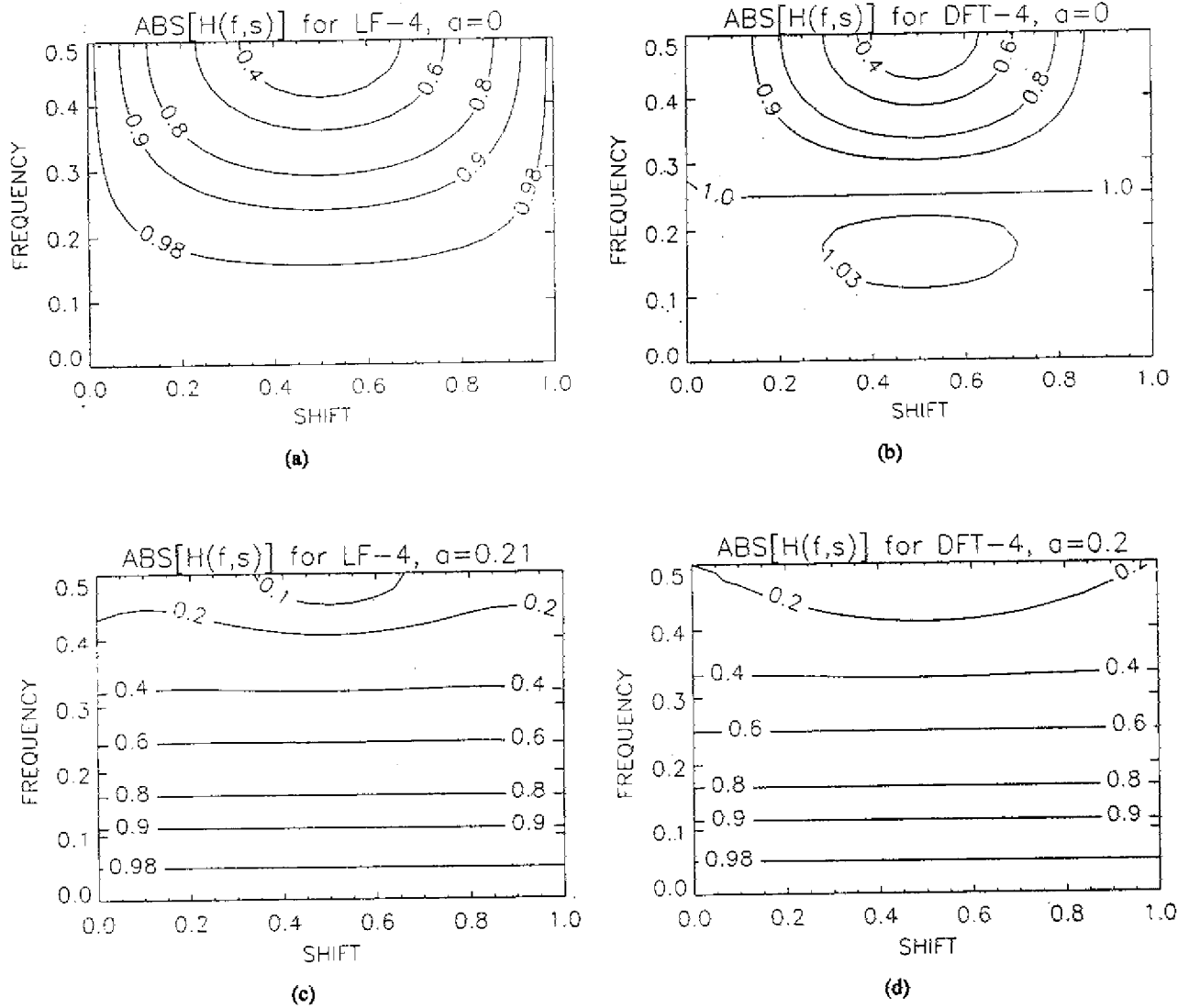
5

Fig. 3 — Contour plots of the amplitude of $H(f,s)$ for LF-4 and DFT-4 with $a = 0$ and a equals best-filtering value. For the latter, the contours are remarkably close to being straight lines parallel to the shift axis, except for frequencies near $f = 1/2$.

For LF-4, the desired value is $a = 0.208 \approx 0.21$, and for DFT-4, it is $a = 0.201 \approx 0.2$, which were found by requiring that

$$\frac{d^2}{ds^2} |H(f_o,s)| = 0 \quad \text{for } s = 1/2, f_o \text{ small.} \tag{12}$$

In Eq. (12), "f_o small" for DFT-4 means that $\cos(2\pi m f_o)$ is adequately approximated by $1 - (2\pi m f_o)^2/2$; then the solution to Eq. (12) is independent of f_o. For LF-4, the expansion of the cosine must be carried to fourth order in f; it turns out that the sum of all the second order terms is zero, regardless of the value of "a." Vanishing of the second derivative is required in Eq. (12) because symmetry of the functional dependence on s means that the first derivative is necessarily zero at $s = 1/2$. Figures 3(c) and (d) show that the contour lines of $|H(f,s)|$ are now almost (but not exactly) straight lines, except for high frequencies. That is, $|H(f,s)|$ is almost completely independent of shift for the low frequencies that dominate most real-world image data. This is more true for DFT-4 than for LF-4; it should therefore be a better interpolator for data that contain more high frequencies, or equivalently, for data that are more sparsely sampled.

6

If the contour lines are very close to straight for most frequencies, we can reasonably expect the same parameter value to work for a wide variety of applications, because the functioning of the interpolator will be nearly independent of the frequency content of the data and of the shift. However, for data that are dominated by high frequencies, it may be desirable to make the high-frequency contours straighter than shown in Figs. 3(c) and (d). This can be done by increasing "a" slightly, to about 0.22 or 0.24, at the cost of having contours that are not straight (slightly convex upward) at lower frequencies. In Sections 6 and 7, we will see examples of interpolators that cannot, with any choice of parameters, achieve the degree of contour line straightness that LF-4 and DFT-4 do. With such nonstraight contours, parameters may need to be adjusted for different applications, i.e., for variable frequency content of the data and for different shifts.

That filtering interpolators can be constructed with nearly straight contours is of particular interest for higher-order frame differencing. A second-order frame differencer might, for example, start with three frames, subtract the second from the first, the third from the second, and then difference the two resulting first-differenced frames. If, as is usually the case, the three frames are not coaligned, interpolations must be performed at each stage of the process. Clutter leakage can be minimized by initially arranging the frames in order of increasing shift (recall that shift is always between zero and one; this may require reindexing the samples of one or more frames), applying a filtering interpolator with zero shift to the right-most (cf, Fig. 1), and applying the same interpolator with appropriate shifts to the other two. With all three frames thus resampled on a common grid and nearly identical spectral filtering applied to each, the second-order difference can be generated without further concerns about registration and with minimal effects from interpolation error. Obviously, the same principle applies to any number of frames or order of differencing, as well as to any other signal processor that compares multiple frames of data.

The assumption that the right-most data are the no-shift data is, of course, purely a matter of convenience. The left-most data could equally well be chosen: the sole effect of this choice would be to reverse the order of the tap weights applied to the shift data. Thus, if several sets of data are involved, any one can be taken to be the unshifted data, with the filtering interpolators derived here applied directly to those shifted to the left and applied with reversed tap weights to those shifted to the right.

The previous discussion has concentrated on the amplitude of $H(f,s)$ rather than its phase. The effect of the phase of a digital filter is to shift the Fourier components of the signal: if the phase of $H(f_1,s)$ is 0.2 rad, a sinusoid of frequency f_1 is shifted by $0.2/2\pi$ cycles. Thus, the phase of the filter and the shift of the signal are equivalent concepts, and the reason that we need not consider the phase of $H(f,s)$ explicitly is that its phase requirements are built into Eq. (5) in the form of the requirement that basis functions be exactly shifted. This point will not be pursued further here, except to exhibit Figs. 4(a) through (d), which show contour plots of the phase errors for LF-4 and DFT-4. The phase error is the difference between the actual phase of the filter and the phase of a perfect shifter (e.g., the Nyquist reconstruction formula). The phase of a perfect shifter is a linear function of frequency with slope $2\pi s$: $\phi = 2\pi sf$. Then each frequency component of the signal is shifted by a distance ($\phi/2\pi$ cycles) × ($1/f$ per cycle) = s, i.e., shift is independent of frequency. In Fig. 4, the phase error is always zero at a shift of 0.5, because of the symmetry of the tapweights [9], which is $A_{-1}(0.5) = A_2(0.5)$, $A_0(0.5) = A_1(0.5)$. In Figs. 4(b) and (d), the phase error is zero at a frequency of 0.25 because DFT-4 interpolates this frequency exactly.

4. TESTING THE INTERPOLATORS

As a first step to seeing how well the interpolators work in practice, they were applied to a representative point spread function (PSF). A PSF is used as a test waveform because it is the most rapidly varying signal that will occur in most data, and, therefore, the hardest to interpolate. (For an
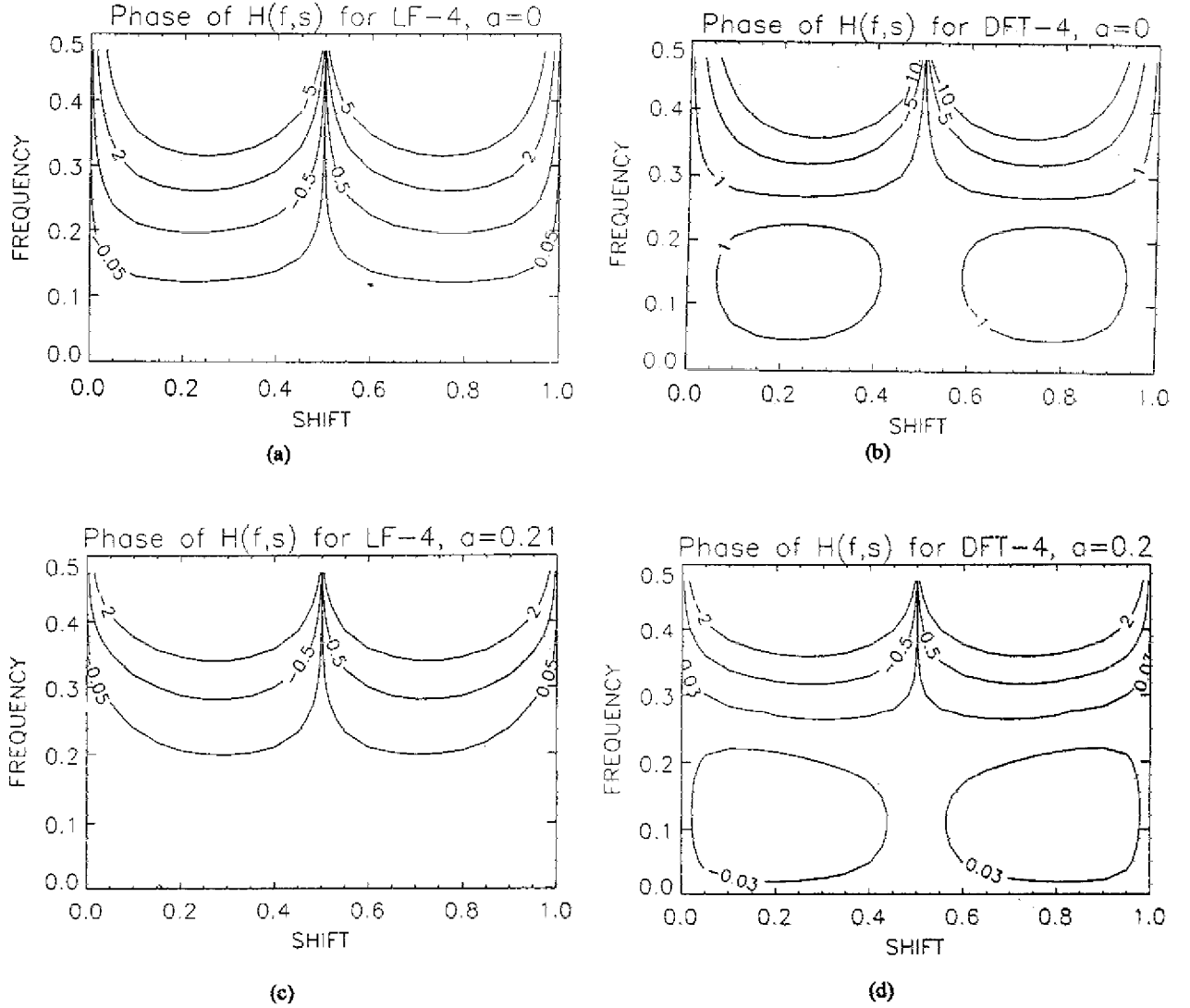
Fig. 4 — Contour plots of the difference, in degrees, between the phase of $H(f,s)$ and the phase of a perfect shifter, for LF-4 and DFT-4

example of data that vary more rapidly, consider a scene containing a positive-contrast point source located very close to a negative-contrast point source; fortunately, such occurrences are rare.) The representative PSF chosen is a unit-height Gaussian, with standard deviation σ given in units of the sample spacing. By changing σ, we can see how well an interpolator works as a function of sampling rate.

Figures 5(a) and (b) show results in which "worst-phased" sampling (samples miss the peak of the curve by the maximum amount) is assumed because it is bound to occur at many points in real data. A standard deviation of $\sigma = 0.8$ is shown because it is a stressing case: it is about the lowest sampling rate for which the interpolators perform well for any value of "a." Also with slightly less than two samples per full width, half maximum (FWHM) of the PSF, it is about as low a sampling rate as a reasonable system should have ($\sigma \approx 1$ or higher gives substantially smaller interpolation errors). For $a = 0$, the long-dashed curve of Fig. 5 (the result of applying the interpolator to the no-shift data) is coincident with the solid curve, and the difference between it and the short-dashed curve (the result of applying the interpolator to the shift data) is substantial. For best-filtering "a," the maximum difference between the
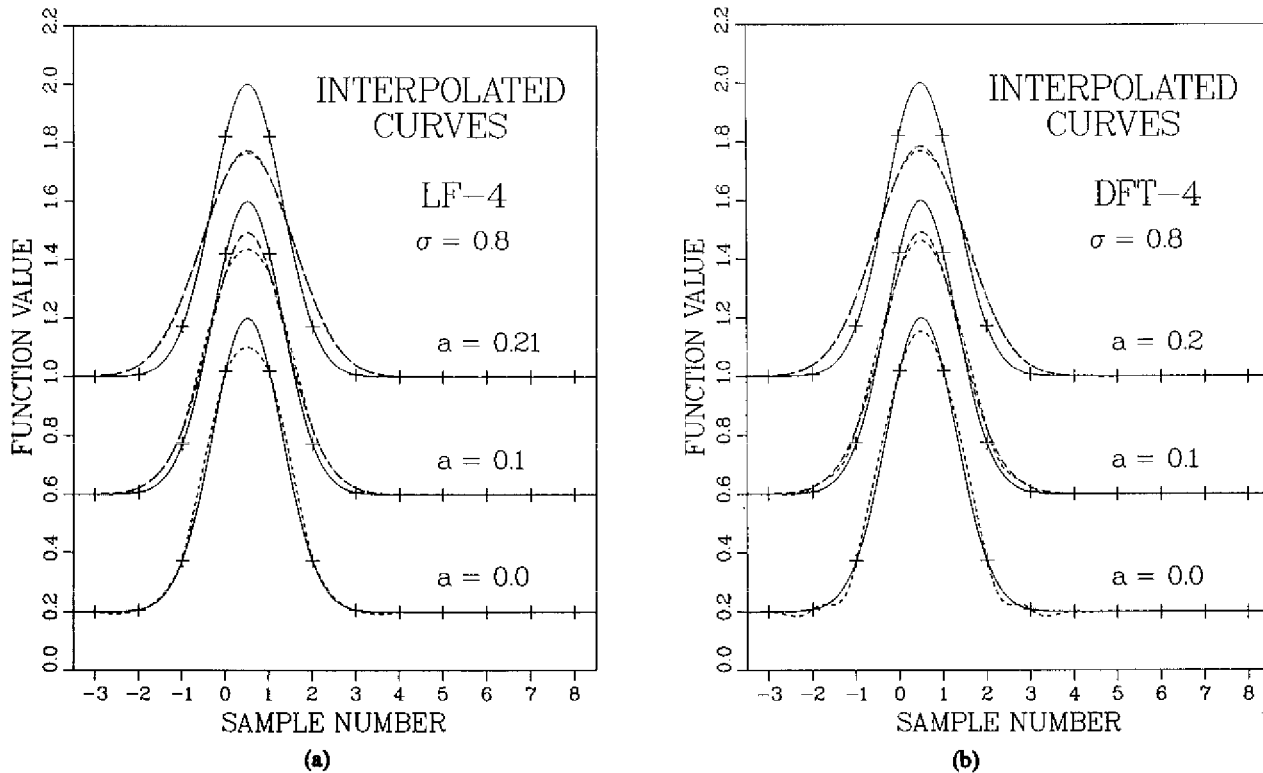
Fig. 5 — Interpolated curves for LF-4 (a) and DFT-4 (b). The solid curves represent Gaussian PSFs, which have been sampled at the +'s. The short-dashed curves show the values interpolated from these samples. The long-dashed curves show the effect of the interpolator on the no-shift data, for all possible no-shift data sets (i.e., on the solid curve). For a = 0, the long-dashed curve is coincident with the solid curve.

long-dashed and short-dashed curves is reduced by about an order of magnitude; $a = 0.1$ is an intermediate case. For best-phased sampling (a sample occurs at the peak of the curve), fits are much better than those of Fig. 5 for $a = 0$ but not noticeably different for best-filtering "a." For $\sigma \geq 1$ and best-filtering "a," the two dashed curves are nearly identical, regardless of sample phasing.

Direct testing of interpolators on real data is very desirable, but seldom possible: an interpolated value can be calculated, but the true value to which it should be compared is usually not available. However, the relative performance of different interpolators can be compared by resampling and differencing simulated imagery and calculating the root mean square (rms) clutter levels of the difference frames. With computer-simulated imagery, we have the luxury of studying interpolation error in isolation, separated from noise, hardware problems such as line-of-sight jitter, or other complications. Tests on imagery of the Earth's surface [10] have verified that the best-filtering a's found above are also the best values to use on imagery. Tests also show that target-to-clutter ratio improvements of a factor of five (14 dB) or more are obtainable, compared to $a = 0$ on highly structured scenes, including an areal view of a city. Improvements on low-structure scenes are similar, but this is probably not important because the dominant source of leakage through an FDSP for these scenes is noise, not interpolation error.

But Figs. 3 and 5 show that the price of this additional clutter suppression is a loss of high-frequency detail. For $a = 0$, only the shift data are filtered and the loss of detail is minimized. For best-filtering "a," both data sets are filtered, and the loss of high frequencies can be substantial. Equations (7) and (8) constitute one-parameter families of interpolators: the user can choose the parameter "a" anywhere between 0 (minimum high-frequency loss, maximum clutter leakage) and its best-filtering value (greatest high-frequency loss, least clutter leakage), depending on the particular target strengths and clutter and noise levels of interest. If, for example, $a = 0.1$ reduces clutter leakage below the level of sensor noise, then the primary effect of further increasing "a" will be to reduce target signature.

9

5. THEORETICAL PERFORMANCE ANALYSIS

There are two reasons why filtering interpolators work better than standard interpolators in preparing data for an FDSP. As emphasized above, the first is that similar spectral filtering is done on *both* data sets (standard interpolators operate on only one data set). The second is that filtering interpolators, more than standard interpolators, reduce the data's high frequency content. This includes the out-of-band contributions from the displaced replicas of the spectrum of the original analog signal that exist in the spectrum of the sampled data and are aliased into the passband by the resampling process. This latter point is addressed in the Appendix, which gives a general analysis of FDSP performance for an arbitrary interpolation function, or kernel, and shows how the kernel acts as a filter. Figure 6 shows the frequency responses of the standard and best-filtering interpolation kernels for LF-4 and DFT-4 (the plots were obtained by Fourier transforming the relevant four-point kernels, which are shown in Fig. 2).

Figure 6 shows that the reductions in the peak sidelobe level obtained with the best-filtering kernels are large: 15 dB for LF-4 and 30 dB for DFT-4. Differences in stopband behavior between the two types of kernels are also instructive. The LF-4 kernels have comparatively high sidelobes with wide nulls around integer multiples of the sampling frequency (i.e., covering the repetitions of low frequencies). The DFT-4 kernels have much narrower nulls but also have a much lower average level for the stopband sidelobes. On this basis, one would expect LF-4 to perform well on low-frequency data, while DFT-4 would be a better choice for data with greater high-frequency content.
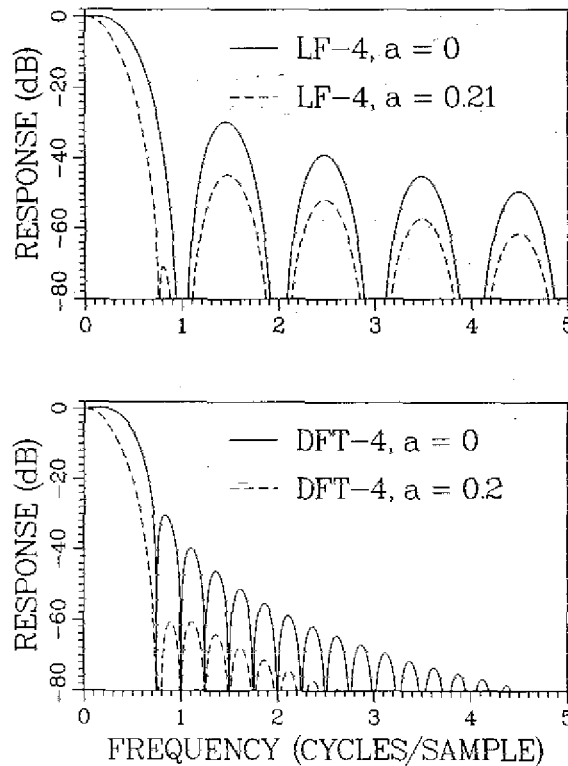

Fig. 6 — The frequency responses of LF-4 and DFT-4 for *a* = 0 and *a* equals best-filtering value. The plotted quantity is twenty times the logarithm of the magnitude of the Fourier transform of the interpolation function.

Of course, the cost of the stopband improvement obtained with filtering interpolators is additional in-band attenuation of the target signal, but the tradeoff is very beneficial in terms of signal-to-clutter ratio (SCR). The Appendix shows how the clutter power leakage and point target response of an FDSP can be expressed as integrals over frequency of the scene and point response power spectra, combined with the frequency response of the interpolation process (Eqs. (A-8) and (A-10)). These results were used to calculate interpolator performances for the specific case of a Gaussian PSF ($\sigma = 0.8$) and the representative background power spectrum given in Eq. (A-9). With the target and background both scaled to unit integrated power, the difference frame SCR given in Eq. (A-11) is a good measure of the signal processing gain of an FDSP for different interpolators, compared to an SCR of unity for the original imagery. Figure 7 shows the shift-dependent SCR gains for LF-4 and DFT-4.
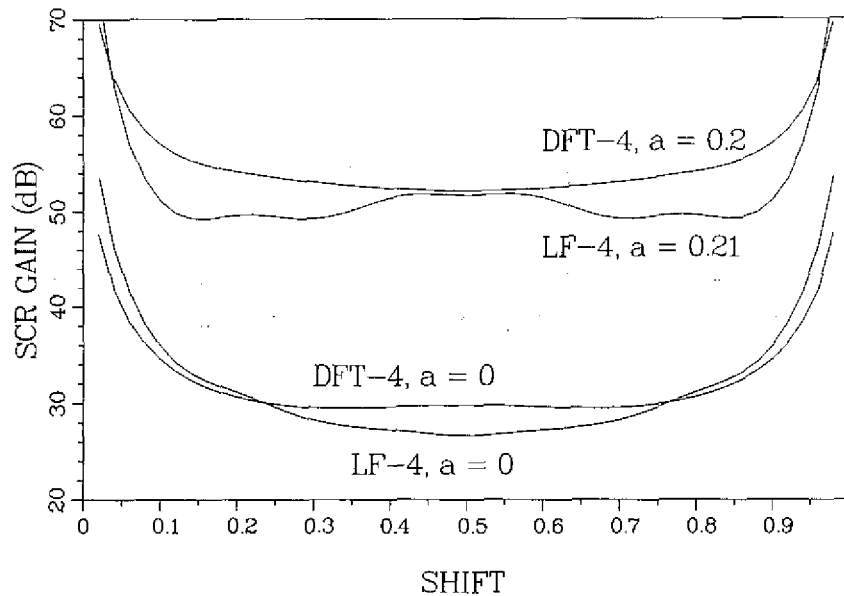

Fig. 7 — Relative interpolation performance found from Eq. (A11) of LF-4 and DFT-4 with $a = 0$ and a equals best-filtering value. Note that worst performance is not necessarily obtained at a shift of 0.5.

The gain shown in Fig. 7 depends on two factors: the clutter reduction in the difference frame and the target attenuation due to no-shift frame filtering. The target attenuation penalties incurred with the best-filtering kernels are overwhelmed by the dramatic improvements in clutter subtraction. For the case considered, SCR improvements of, typically, about 20 dB (a factor of 10 in amplitude) are obtained by using optimum filtering interpolators in lieu of standard interpolators. Note that, except for shifts near zero or one, the best-filtering DFT-4 kernel yields up to 6 dB more gain than the best-filtering LF-4 kernel (for the model spectrum assumed). Figure 7 also shows the somewhat counterintuitive fact that a shift of 1/2 is not always the worst case for interpolation error.

The reduction in target signature found by evaluating the integral in Eq. (A-10) with and without the filtering term in the integrand is about 1 dB for best-filtering "a," for the case considered here (Gaussian PSF, $\sigma = 0.8$). This result is for integrated signal power. For many systems, the reduction in peak target signal is more important; it can be seen in Fig. 5 to be about 2 dB, or a factor of about 0.8 in amplitude. The maximum possible reduction of peak target signal, which occurs for a target that has only one sample on it, can be seen from Eq. (6) to be $1 - 2a$; ordinarily, data are sampled often enough that this degree of target signature reduction will not occur.

The typical improvement of 20 dB given above is better than the 14 dB improvement quoted in Section 4 for highly structured imagery, partly because of the particular model background power spectrum used here and partly because only a one-dimensional (1-D) problem is being considered. For isotropic two-dimensional (2-D) data, errors resulting from interpolation in one direction are statistically independent of errors resulting from interpolation in the other; the two components are added in quadrature to find total clutter. Thus, if a filtering interpolator reduces one component of clutter by some factor, it should reduce the other component, and, consequently, the total clutter, by the same factor; i.e., clutter reduction is independent of dimensionality. But target signature reduction is essentially multiplicative: if a 1-D interpolation reduces a target signal by 2 dB, then a 2-D interpolation will reduce it by (about) 4 dB. Thus, for properly related 1-D and 2-D power spectra [11] (i.e., both are obtained from the same scene), SCR gain will be lower for 2-D data than for 1-D data.

The effect of the interpolation process on the statistical properties of the noise can be evaluated by choosing a data set $\{x_n\}$ that consists of noise with zero mean and variance $\langle x^2 \rangle = \sigma^2$. It is easy to show from Eq. (1) that $\langle y_s \rangle = 0$ and that

$$\langle y_s^2 \rangle = \sum_{n=-(N/2-1)}^{N/2} \sigma^2 A_n^2(s) + 2 \sum_{m=1}^{N-1} \sum_{n=-(N/2-1)}^{N/2-m} \langle x_n x_{n+m} \rangle A_n(s) A_{n+m}(s), \tag{13}$$

i.e., the variance of the interpolated value depends on the autocorrelation function of the noise. For uncorrelated noise, only the first sum contributes; Fig. 8 plots the result.
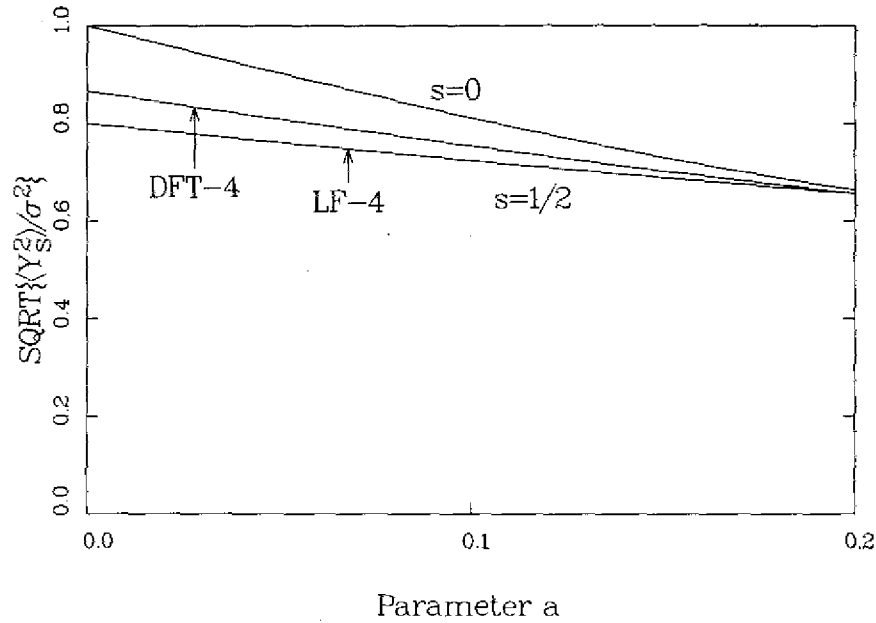

Parameter a

Fig. 8 — The curves show the effect of the interpolators on the standard deviation of uncorrelated noise, as a function of the parameter "a," from Eq. (13). For $s = 0$ (top curve), LF-4 and DFT-4 do the same thing. The greatest amount of noise filtering, shown in the lower curves, occurs for $s = 1/2$.

6. MORE INTERPOLATORS/POSSIBLE SPLINE APPLICATION

Equations (7) and (8) show that the parts of the A_n that depend on the parameter "a" can be separated out. Denoting this part, which is the difference between standard ($a = 0$) and filtering ($a \neq 0$) interpolators, by ΔA_n, we have

$$\Delta A_{-1} = a - as,$$

$$\Delta A_0(s) = 2a + 3as,$$

$$\Delta A_1(s) = a - 3as, \tag{14}$$

$$\Delta A_2(s) = as,$$

for LF-4. These ΔA's can be added to any standard $N = 4$ interpolator to make a filtering interpolator, and the same can be said for the ΔA's obtained from Eq. (8). Interpolators generated in this manner no longer have the property that they interpolate a simple set of N-basis functions exactly (which is also true of most standard interpolators). Another way to generate new interpolators is to use other sets of basis functions in Eq. 5.

For example, Eq. (14) can be added to the coefficients of the parameterized cubic interpolator described by Wolberg [2] to yield

$$A_{-1}(s) = -4\alpha + 8\alpha(1 + s) - 5\alpha (1 + s)^2 + \alpha (1 + s)^3 + a - as,$$

$$A_0(s) = 1 - (\alpha + 3) s^2 + (\alpha + 2)s^3 \qquad\qquad - 2a + 3as,$$

$$A_1(s) = 1 - (\alpha + 3)(1 - s)^2 + (\alpha + 2)(1 - s)^3 \qquad + a - 3as, \tag{15}$$

$$A_2(s) = -4\alpha + 8\alpha (2 - s) - 5\alpha (2 - s)^2 + a (2 - s)^3 + as.$$

We take the parameter α (which alters the frequency response of the interpolator) to be $-1/2$ (the "image independent optimal choice" of Park and Schowengerdt [12]). For this interpolator, it turns out that no choice of "a" makes the contour lines of $|H(f,s)|$, shown in Fig. 9, as flat as those for LF-4 and DFT-4, except for frequencies very near $f = 0$. By trial and error on simulated imagery, we found that (coincidentally) $a = 0.2$ works best. As can be deduced from Fig. 9, the problem with this interpolator is that it works well (as well as LF-4) for shifts near zero and one-half, but rather poorly (though still better than its standard form) for shifts near one-quarter or three-quarters. Other values of α and "a" may work for other data, but the user must proceed at his own risk.

Another candidate for modification is the kernel for cubic B-spline interpolation [2], which is used here as a local rather than a spline interpolator. (Pratt [13] also uses it this way but does not apply it to the frame-differencing problem.) In its basic form, $A_{-1}(0) = A_1(0) = 1/6$ and $A_0(0) = 2/3$, so it is clearly a filtering interpolator; in fact, it is the first one we used to test the idea. The interpolation coefficients are
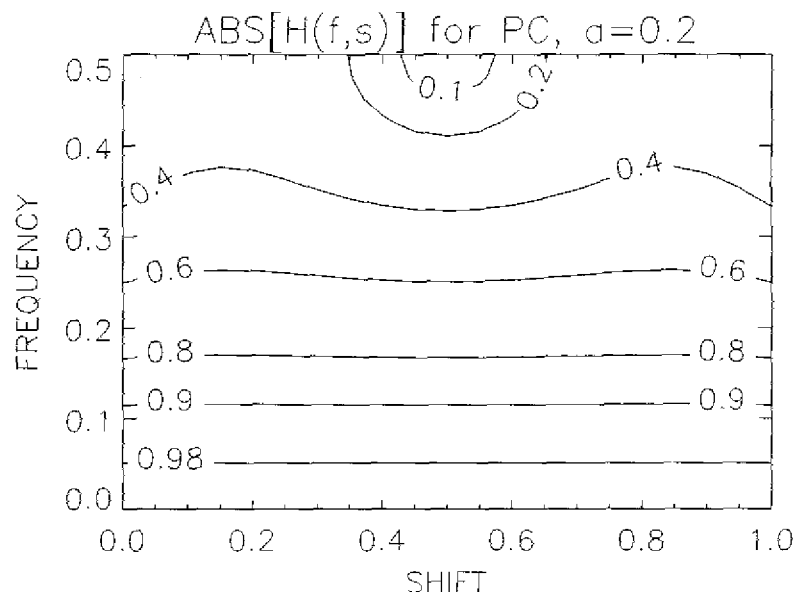
Fig. 9 — Contour plot for parameterized cubic (PC) interpolation. Most contours have the same value near $s = 1/2$ as near $s = 0$. Hence this interpolator should work well for shifts near 1/2, but not for shifts near 1/4 or 3/4.

$$A_{-1}(s) = \frac{1}{6} [1 - 3s + 3s^2 - s^3],$$

$$A_0(s) = \frac{1}{6} [4 - 6s^2 + 3s^3],$$

$$A_1(s) = \frac{1}{6} [1 + 3s + 3s^2 - 3s^3], \qquad (16)$$

$$A_2(s) = \frac{1}{6} s^3.$$

We could add Eq. (14) to these to change their filtering properties, but inspection of Eqs. (7) shows that Eqs. (16) are equal to LF-4 with $a = 1/6$; i.e., the cubic B-spline interpolation kernel is a special case of parameterized LF-4.

The DFT-4 interpolation kernel developed here can also be applied to spline interpolation. A four-point spline interpolation function $h(x)$ should be (a) everywhere nonnegative and (b) monotonically decreasing to zero for $0 \le x \le 2$ (cf. Fig. 2 and Ref. [2]). The only value for which LF-4 has these properties is $a = 1/6$, but DFT-4 has them for $1/6 \le a \le 3/10$. We suggest, therefore, that DFT-4 with "a" in this range should be an interesting candidate as a kernel for spline interpolation.

7. HIGHER-ORDER INTERPOLATORS

For some applications, higher-order interpolators may be desirable. Lucke [6] gives a formula for standard DFT interpolators of arbitrary order, and, of course, the standard LF-N interpolators are available in many software packages. Generating filtering interpolators from LF-N or DFT-N for $N > 4$ increases rapidly in difficulty because of the number of equations that must be solved and the number of parameters in the no-shift filter. Results for $N = 6$ will be given.

For six-point interpolators, imposing the same conditions as for Eqs. (6), the $A_n(0)$ are given by two parameters, a and b:

$$A_{-2}(0) = b, \qquad\qquad A_1(0) = a,$$

$$A_{-1}(0) = a, \qquad\qquad A_2(0) = b, \qquad\qquad (17)$$

$$A_0(0) = 1 - 2a - 2b, \qquad A_3(0) = 0.$$

With two parameters, we can now place two conditions on the contours of $|H(f,s)|$. We could, for example, extend the criterion of Eq. (12) to the fourth derivative to obtain maximal straightness of the contour lines near $f = 0$ (condition 1). Or we could meet the criterion of Eq. (12) at two different frequencies (condition 2).

For LF-6, the interpolation coefficients are

$$A_1(s) = \frac{1}{12} \{12a + 4[3 - 2(4a + b)]s + 8[1 - 3(a + 4b)]s^2 - [7 - 20(a + 4b)]s^3 - 2s^4 + s^5\},$$

$$A_2(s) = \frac{1}{24} \{24b - 2[3 - 4(4a + b)]s - [1 - 12(a + 4b)]s^2 + [7 - 20(a + 4b)]s^3 + s^4 - s^5\},$$
$$\qquad\qquad (18)$$

$$A_3(s) = \frac{1}{120} \{4[1 - 5(a - 2b)]s - 5[1 - 4(a + 4b)]s^3 + s^5\},$$

$$A_{-2}(s) = A_3(1 - s), \quad A_{-1}(s) = A_2(1 - s), \quad A_0(s) = A_1(1 - s).$$

With this interpolator, as for the parameterized cubic interpolator of Section 6, the parameters that fulfill condition 1, namely $a = 0.193$, $b = 0.101$, do not give sufficiently straight contours for larger f. The parameters that fulfill condition 2, with f_o small and $f_o = 1/4$, namely $a = 0.255$, $b = 0.068$, do not work much better and result in a fairly severe loss of high frequency detail. Once again, we find that no choice of parameters gives really straight contours. Hence, for this interpolator, we should expect that different applications will require different parameter values. Good candidate parameter sets to start with are those which give the best fits for our Gaussian PSFs, $a = 0.24$ and $b = 0$ (shown in Fig. 10(a)) and those which give the best results on our simulated imagery, $a = 0.26$ and $b = 0.05$.

For DFT-6, the basis functions are

$$f_1(x) = 1, \qquad\qquad f_4(x) = \cos(2\pi x/3),$$

$$f_2(x) = \cos(\pi x/3), \qquad f_5(x) = \sin(2\pi x/3), \qquad (19)$$

$$f_3(x) = \sin(\pi x/3), \qquad f_6(x) = \cos(\pi x).$$

The resulting interpolation function is

$$h(x) = \frac{1}{6} [1 + 2(1 - a - 3b)\cos(\pi x/3) + 2(1 - 3a - 3b)\cos(2\pi x/3)$$
$$\qquad\qquad (20)$$
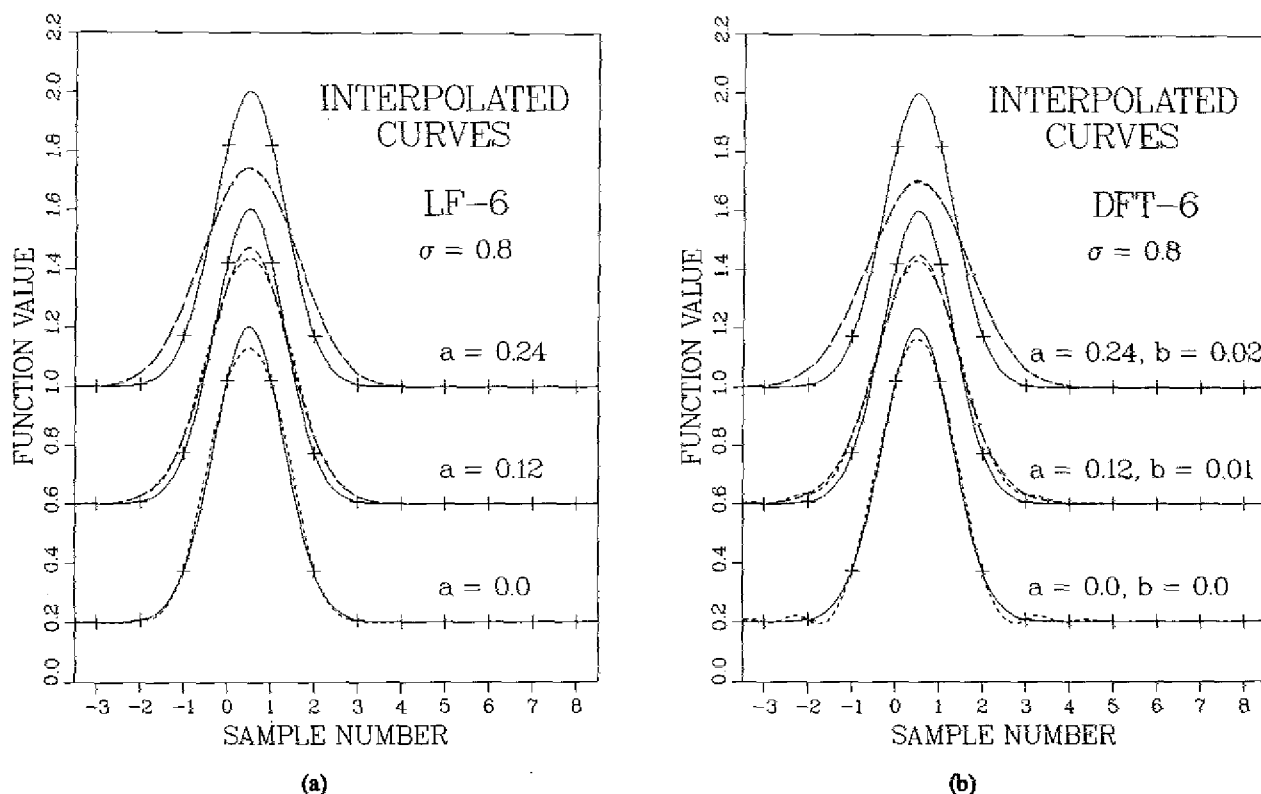$$+ (1 - 4a)\cos(\pi x)]$$

Fig. 10 — Interpolated curves for LF-6 (a) and DFT-6 (b).
For LF-6, $b = 0$ for all curves.

for $|x| \leq 3$ and zero otherwise, and $A_n(s) = h(n - s)$, $n = -2, -1, 0, 1, 2, 3$. The parameter values found by imposing condition 1 are $a = 0.236 \approx 0.24$, $b = 0.019 \approx 0.02$. In this case, the contour lines of $|H(f,s)|$ are quite straight except for frequencies close to $f = 1/2$, and these parameters work well on the Gaussian PSFs, as shown in Fig. 10(b), and on simulated imagery. For $\sigma = 0.8$, this interpolator, in its best-filtering form, produces errors about a factor of two smaller than DFT-4, as can be seen by comparing Fig. 10(b) with Fig. 5(b). At lower sampling rates, DFT-6 maintains a performance advantage over DFT-4, but the sampling rate cannot decrease much before the data become so poorly sampled that no interpolator works well. At a slightly higher sampling rate ($\sigma \geq 1$), the performance difference between DFT-4 and 6 is much reduced: for well-sampled data, any interpolator gives accurate results.

8. CONCLUSIONS

We think that by far the most useful interpolator given here is DFT-4 with $a = 0.2$ (best-filtering "a"). We recommend it for general applications, even if the resulting interpolation error is substantially smaller than the noise. LF-4 with $a = 0.21$ is nearly the same, but produces slightly more high-frequency loss, and, as can be seen from its interpolation function (Fig. 2), creates (very) small discontinuities in the first derivative of the interpolated data. Only if minimizing high frequency losses is a major concern is it likely to be worthwhile adjusting "a" to a smaller value, which adjustment (for best results) must be redone for data sets with different spectral content. Only for extremely low-noise data would a higher-order interpolator be beneficial.

If the best-filtering value cannot be used, i.e., "a" is in the range of, say, $0 \leq a \leq 0.15$, the choice between LF-4 and DFT-4 is determined by the frequency content of the data: the difference will usually not be large, but LF-4 can be expected to give better results on data that are more strongly dominated

by low frequencies (or are more densely sampled). Six-point interpolators will improve performance in this case, but we have scant experience on which to base recommendations. Inspection of Fig. 10 indicates that $b = 0$, $a \leq 0.1$ should work fairly well; if $a > 0.1$ must be used, probably DFT-4 with $a = 0.2$ would be a better (or at least a simpler) choice.

For special applications, Section 6 gives instructions on putting the interpolator of the user's choice into filtering form. The filtering interpolators introduced here are of the local, convolutional type but also lead to a candidate kernel for cubic spline interpolation (Section 6 again). The new interpolators are not appropriate for data with high sampling rates, because well-sampled data can be interpolated with high accuracy by standard interpolators; in such applications, the primary effect of the new interpolators would be to reduce target signatures. But for moderate-to-low sampling rates, we have found SCR improvements of a factor of five or more for point targets against highly structured scenes, when interpolation error is the dominant source of clutter. In our experience with simulations of a state-of-the-art IR surveillance sensor, clutter leakage due to interpolation error can always be brought below the level resulting from other effects, such as line-of-sight jitter, inaccuracies in determining the shift, calibration errors, and detector noise.

REFERENCES

[1] W.T. Press, et. al., *Numerical Recipes* (Cambridge University Press, New York, 1986), Ch. 3, pp. 80-82.

[2] G. Wolberg, *Digital Image Warping* (IEEE Computer Society Press, Los Alamitos, CA, 1990), Ch. 5, pp. 130-135.

[3] C. Kuglin and D. Hines, *The Phase Correlation Image Alignment Method*, IEEE Proceedings of the Conference on Cybernetics and Society, San Francisco, CA, Sept. 23-25, 1975, pp. 163-165.

[4] A. Schaum and M. McHugh, "Analytic Methods of Image Registration: Displacement Estimation and Resampling," NRL Report 9298, Feb. 1991.

[5] W. Shaffer and R. Lucke, "Simultaneous Registration and Nonuniformity Correction of Space-Based IR Images for a Scanning Sensor," NRL Report 9390, May 1992.

[6] R.L. Lucke, "A Local Interpolator Derived from the Discrete Fourier Transform," NRL Report 9392, Apr. 1992.

[7] A. Schaum, "Dual Difference Filtering for Change Detection in Digital Signals," submitted for publication review.

[8] A.V. Oppenheim and R.W. Schafer, *Digital Signal Processing* (Prentice-Hall, Inc., Englewood, NJ, 1975), p. 238.

[9] *Ibid*, p. 157.

[10] R. Lucke, W. Shaffer, and R. Rhodes, "Computer Simulation of Space-Based Infrared Surveillance," NRL Report 9508, in preparation.

[11] D.L. Hench, "Clutter Leakage Analysis for Satellite-Based Aircraft Detection Using IR Sensors," TR-944, Optical Sciences Company, Placentia, CA, pp. 20-22 (1988).

[12] S.K. Park, and R.A. Schowengerdt, "Computer Vision, Graphics, and Image Processing," **23**, pp. 258-272 (1983).

[13] W.K. Pratt, *Digital Image Processing* (John Wiley & Sons, Inc., New York, 1978), pp. 113-14.

Appendix

FOURIER ANALYSIS OF INTRERPOLATORS
FOR FRAME DIFFERENCING

In the following analysis, a continuous signal is first reconstructed from the sampled data by convolving with an interpolation kernel; the result is then resampled at the specified shift. This interpretation of the resampling problem will show explicitly how the interpolation function, or kernel, acts as a filter on the data. Since any finite interpolation filter has a frequency response of infinite extent, it invariably passes out-of-band components in the Fourier transform of the sampled data. These out-of-band components alias into the base-band when the signal is resampled. Expressions for the resulting power in a difference frame will be derived for a representative background clutter spectrum and for a target spectrum. The one-dimensional case is done here; the results are easily extended to two dimensions.

Let $y(x)$ be a continuous bandlimited signal with Fourier transform $Y(f)$ defined on the normalized frequency band $-0.5 \le f \le 0.5$. This signal is present in two distinct frames and is sampled at equally spaced but differently phased intervals at the Nyquist rate (i.e., unity sampling interval), as shown in Fig. 1. The discrete frame observations (regarded as functions that are defined everywhere but are nonzero only at the sample points) and their Fourier transforms are expressed by

$$f_1(x) = y(x) \sum_m \delta(x - m), \qquad F_1(f) = \sum_m Y(f - m),$$

$$f_2(x) = y(x) \sum_m \delta(x - m - s), \qquad F_2(f) = \sum_m Y(f - m)e^{-j2\pi sm}, \tag{A1}$$

where m ranges from $-\infty$ to ∞, and s is the shift. Also, $f_1(x)$ and $f_2(x)$ describe the shift and no-shift data of Fig. 1, respectively.

The frames are to be resampled to spatially align the signal prior to differencing. The first step is linear filtering with a (continuous) interpolation kernel with finite impulse response $h(x)$ (this is the interpolation function, cf. Fig. 2) and frequency response $H(f)$ (the magnitude of $H(f)$ for LF-4 and DFT-4 is shown in Fig. 6). This produces a pair of continuous functions denoted by g_1 and g_2:

$$g_1(x) = f_1(x) * h(x), \qquad G_1(f) = H(f) \sum_m Y(f - m),$$

$$g_2(x) = f_2(x) * h(x), \qquad G_2(f) = H(f) \sum_m Y(f - m)e^{-j2\pi sm}. \tag{A2}$$

To obtain a pair of discrete frames that are spatially co-aligned, $g_1(x)$ and $g_2(x)$ are both resampled at $x = n + s$, where n is an integer (see Fig. 1). This represents a shift for the samples from which $g_1(x)$ was derived but not for the samples that generated $g_2(x)$. The resampled discrete frames and their Fourier transforms are given by

$$g_{1R}(x) = g_1(x) \sum_n \delta(x - n - s), \qquad G_{1R}(f) = \sum_n H(f - n) \sum_m Y(f - m - n)e^{-j2\pi sn},$$

$$g_{2R}(x) = g_2(x) \sum_n \delta(x - n - s), \qquad G_{2R}(f) = \sum_n H(f - n) \sum_m Y(f - m - n)e^{-j2\pi s(m+n)}.$$

(A3)

G_{1R} and G_{2R} are defined over $-\infty < f < \infty$, but, since they are the Fourier transforms of sampled data, only their values over $-0.5 \leq f \leq 0.5$ are of interest. Since the signal spectrum $Y(f)$ is assumed to be bandlimited to this range, the only nonzero terms in the summations over m are $m = -n$, when $-0.5 \leq f \leq 0.5$. The resampled frame transforms then simplify to

$$G_{1R}(f) = Y(f) \sum_n H(f - n)e^{-j2\pi sn}, \qquad -0.5 \leq f \leq 0.5,$$

$$G_{2R}(f) = Y(f) \sum_n H(f - n), \qquad -0.5 \leq f \leq 0.5.$$

(A4)

The spectral content of the difference frame is therefore given by

$$D_{12}(f) = G_{2R}(f) - G_{1R}(f) = Y(f) E(f,s),$$

(A5)

where

$$E(f,s) = \sum_n H(f - n)[1 - e^{-j2\pi sn}]$$

(A6)

is a shift-dependent error filter defined for $-0.5 \leq f \leq 0.5$. Thus $E(f,s)$ is a weighted sum of kernel response samples spaced uniformly at intervals of the sampling frequency, with the response in each repetition interval phase-shifted by an amount that depends on the frame shift s. The difference frame is always zero at integer-valued shifts. Note that, for the ideal sinc function interpolator, $H(f) = \text{Rect}(f)$ (i.e., $H(f) = 1$ for $|f| \leq 1/2$, $H(f) = 0$ otherwise), so that $E(f,s) = 0$ and the difference frame in Eq. (A5) is identically zero. In general, however, Eqs. (A5) and (A6) show that the difference frame clutter leakage for a finite interpolator depends on the out-of-band frequency response, the data spectrum, and the shift.

The shift-dependent power due to clutter in the difference frame is

$$P_c(s) = \int_{-0.5}^{0.5} |D_{12}(f)|^2 \, df = \int_{-0.5}^{0.5} |Y(f)|^2 |E(f,s)|^2 \, df.$$

(A7)

This expression has been derived for an arbitrary deterministic signal with Fourier transform $Y(f)$. For a random signal (e.g., clutter or noise), we can use the fact that the expected value of $|Y(f)|^2$ over the ensemble is the discrete signal power spectrum $S(f)$ to obtain the average error power expression

$$\overline{P_c}(s) = EP_c(s) = \int_{-0.5}^{0.5} S(f)|E(f,s)|^2 \, df.$$

(A8)

To evaluate Eq. (A8), we need a representative background clutter power spectrum to use for $S(f)$. We have found the following formula, which gives unit integrated power over the range $-0.5 \leq f \leq 0.5$, to be a useful model for imagery, with n generally in the range of 4 to 6:

$$S(f) = (n + 1)(1 - 2|f|)^n, \qquad -0.5 \leq f \leq 0.5.$$

(A9)

Taking $n = 4$ yields the clutter power model used in Eq. (A8) to calculate the results given in Section 5.

The target response attenuation in the no-shift frame caused by the use of a filtering interpolator must also be calculated to obtain the signal-to-clutter ratio at the frame differencing signal processor's output. Given a frequency-domain description $T(f)$ of the target signal shape, the output target power in the no-shift frame is found from Eq. (A4), with $Y(f)$ replaced by $T(f)$, to be

$$P_T = \int_{-0.5}^{0.5} |T(f)|^2 \ |\sum_n H(f - n)|^2 \ df. \tag{A10}$$

The output target power in the shift frame could be similarly calculated; inspection of Eqs. (A4) and (A10) shows that it is shift-dependent. We now assume that the target moves sufficiently between frames that there is no significant interference of the target waveforms when the difference is taken. With this assumption, Eqs. (A8) and (A10) can be combined to compute the output signal-to-clutter ratio given in Fig. 7:

$$SCR(s) = \frac{P_T}{P_c(s)}. \tag{A11}$$

The results, Eqs. (A8), (A10), and (A11), along with the model power spectrum Eq. (A9), were used to calculate the performance figures for standard and filtering interpolators given in Section 5.

We have attempted to improve interpolation performance by adjusting the parameter "a:" this determines $H(f)$ and the error fitler $E(f,s)$. Another degree of freedom can be developed by dropping the condition that only the first frame is shifted. For example, we might "split the difference" between the two frames by moving one to the right by $s/2$, the other to the left by $s/2$. The latter is the same as a rightward shift by $1 - s/2$. Inspection of Fig. 3 shows that $|H(f,s)|$ is the same function of frequency for a shift of $1 - s/2$ as it is for $s/2$, so a "two-way" shift produces identical (in amplitude) spectral filtering of the two data sets. But it does not necessarily give better interpolation results than a one-way shift. To see why this is so, we first note that if resampling is not to be done on one of the original sampling grids, there is no reason why it must be done at $s/2$; it could be done anywhere in the $[0,s]$ interval. Such a two-way shift corresponds to replacing $x - n - s$ as the argument of the δ functions on the left-hand side of Eq. (A3) by $x - n - \beta s$, with $0 \leq \beta \leq 1$.

We now have another parameter to adjust and a more general error filter:

$$E_2(f,s) = \sum_{n=-\infty}^{\infty} H(f - n)[e^{j2\pi(1-\beta)sn} - e^{-j2\pi\beta sn}]$$

$$= 2j \sum_{n=1}^{\infty} \sin(\pi sn)[H(n - f)e^{j\pi(1-2\beta)sn} - H(n + f)e^{-j\pi(1-2\beta)sn}], \tag{A12}$$

where the fact that $H(f)$ is an even function (it is the Fourier transform of $h(x)$, which is real and even) has been used. Next, we rewrite Eq. (A6) in the form

$$E_1(f,s) = \sum_{n=-\infty}^{\infty} H(f - n)[1 - e^{-j2\pi sn}]$$

$$= 2j \sum_{n=1}^{\infty} \sin(\pi sn)[H(n - f)e^{-j\pi sn} - H(n + f)e^{j\pi sn}]. \tag{A13}$$

The relation between Eqs. (A12) and (A13) is hard to interpret in general, but the special case $\beta = 1/2$, $s = 1/2$ is instructive. Equations (A13) and (A12) become, respectively,

$$E_1(f,1/2) \approx 2[H(1 - f) + H(1 + f)], \tag{A14}$$

and

$$E_2(f,1/2) \approx 2j[H(1 - f) - H(1 + f)], \tag{A15}$$

where the approximation comes from the fact that the $n = 1$ term dominates the sum (because only odd terms give nonzero contributions, which decrease rapidly as n increases, as shown in Fig. 6).

Equations (A14) and (A15) show that the relative performance of one-way (E_1) and two-way (E_2) interpolations depends on the relative signs and magnitudes of H just above and below unit frequency. For example, the relevant sidelobes for DFT-4 (see Fig. 6) have an opposite sign before the absolute value is taken, hence tend to cancel in Eq. (A14) and reinforce in Eq. (A15) (i.e., $|H(1 - f) + H(1 + f)| < |H(1 - f) - H(1 + f)|$ for $0 \le f \le 0.25$). Recalling that imagery tends to be dominated by low frequencies, we therefore expect that a one-way shift will give a smaller error than a two-way shift, at least for $\beta, s = 1/2$. Trials on our Gaussian point spread functions and simulated imagery and evaluation of Eq. (A11) bear out this expectation.

In general, β could be chosen to minimize Eqs. (A7) or (A8) for a particular interpolator, signal power spectrum, and shift. We have not investigated this point extensively, but have found improvements in clutter suppression for "a" near zero to be much less than can be obtained by using best-filtering "a." For best-filtering "a," improvements in clutter suppression are likely to be significant only for very low-noise data (otherwise, using the best-filtering value of "a" will, by itself, reduce interpolation error below the noise level). For DFT-4 with best-filtering "a," we have found $\beta = 1$ (conventional one-way interpolation) to give the smallest interpolation errors. For LF-4 with best-filtering "a," a two-way shift with $\beta = 1/2$ tends to give better results than one-way when $s \le 1/4$.